

Lazy global feedbacks for quantized nonlinear event systems

Stefan Jerg, Oliver Junge[‡]

March 2012

Abstract

We consider nonlinear event systems with quantized state information and design a globally stabilizing controller from which only the minimal required number of control value changes along the feedback trajectory to a given initial condition is transmitted to the plant. In addition, we present a non-optimal heuristic approach which might reduce the number of control value changes and requires a lower computational effort. The constructions are illustrated by two numerical examples.

1 Introduction

Traditionally, controllers for (nonlinear) systems have been designed using a continuum as the underlying time domain. With the rise of digital information processing, *time-triggered* or *sampled-data* controller designs have become popular. There, a regular grid of time instances serves as the time domain, cf. [AW97]. Both schemes close the control loop independently of the system's behavior. This might lead to unnecessary communication between the controller and the plant. In the case that the communication is implemented via a digital network, restrictions like the maximal bandwidth or load dependent stochastic effects might play a role and influence the behavior of the closed loop system. In order to decrease the network load and possibly avoid these effects, in *event based* control, information is only transmitted when necessary in order to ensure stability of the closed loop system, see, e.g., [Arz99, OMT02, AB02, TW06, KB06, VK06, HSVDB07, Ast08, GJ08, LL09].

*Faculty for Mathematics, Technische Universität München, 85748 Garching, Germany, jerg@ma.tum.de, oj@tum.de

[‡]This work was supported in part by the priority program SPP 1305 of the German research foundation (DFG) and by the Bavarian State Ministry of Sciences, Research and Arts through the PhD program TopMath

Another means of reducing the amount of data which has to be transmitted (and thus further reducing the network load) is to use a *quantization* of an underlying continuous state space. While any real number which is transmitted digitally, necessarily comes from a quantized set since only finitely many digits can be transmitted, here one aims for quantizations which are as coarse as possible since then fewer bits will suffice to encode the data. We refer to, e.g., [Hsu92, Lun94, FJL02, Sch03].

Recently, a new approach for the construction of controllers for quantized systems has been proposed which relies on a set oriented approach in combination with graph theoretic techniques, cf. [JO04, GJ07, GJ08]. In [GM09], this approach has been extended to event systems.

In the present contribution we describe how to extend this approach such that the number of times that data has to be transmitted from the controller to the plant is minimized along the feedback trajectory to some initial condition. The construction is based on the optimality principle with a suitably chosen state space and cost function. Additionally, we present a non-optimal heuristic approach which also reduces the number of data transmission events while requiring a significantly lower computational effort. The two constructions are illustrated by two numerical examples, an inverted pendulum and a thermofluid batch process.

The paper is structured as follows: In Section 2 we briefly summarize the basic construction from [GJ08, GM09]. In Section 3 we describe how to extend this construction such that the number of data transmissions from the controller to the plant is minimized and illustrate this construction by two numerical examples in Section 4. Finally, in Section 5, we propose the heuristic scheme for reducing the communication effort. Again, this is illustrated by the two examples from Section 4.

2 Global optimal feedbacks for quantized nonlinear event systems

In this section, we summarize the constructions from [GJ08, GM09]: We are given a plant which is modeled by a nonlinear discrete time control system (which may, e.g., be derived from a continuous time system by time sampling)

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)), \quad k = 0, 1, 2, \dots, \quad (1)$$

where $f : \mathcal{X} \times \mathcal{U} \rightarrow \mathcal{X}$ is continuous, $\mathbf{x}(k) \in \mathcal{X}$ is the state and $\mathbf{u}(k) \in \mathcal{U}$ is the control input, $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{U} \subset \mathbb{R}^m$ compact. In addition to f , we are given a continuous running cost function $c : \mathcal{X} \times \mathcal{U} \rightarrow [0, \infty)$ as well as a *target set* $\mathcal{X}^* \subset \mathcal{X}$. We assume c to satisfy $c(\mathbf{x}, \mathbf{u}) = 0$ iff $\mathbf{x} \in \mathcal{X}^*$. Our goal is to compute a feedback law for this system which drives the system into the target set \mathcal{X}^* while minimizing the

accumulated cost. However, the information which is transmitted from the plant to the controller is restricted in the following two ways:

1. *Event model*: The controller only receives information on the state whenever an *event* occurs. Formally, based on the discrete time model (1) of the plant, we are dealing with the discrete time system

$$\mathbf{x}(\ell + 1) = \tilde{f}(\mathbf{x}(\ell), \mathbf{u}(\ell)), \quad \ell = 0, 1, \dots, \quad (2)$$

where

$$\tilde{f}(\mathbf{x}, \mathbf{u}) = f^{r(\mathbf{x}, \mathbf{u})}(\mathbf{x}, \mathbf{u}), \quad (3)$$

$r : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{N}_0$ is a given *event function* and the iterate f^r is defined by $f^0(\mathbf{x}, \mathbf{u}) = \mathbf{x}$ and $f^r(\mathbf{x}, \mathbf{u}) = f(f^{r-1}(\mathbf{x}, \mathbf{u}), \mathbf{u})$, cf. [GM09]. Accordingly, we define an associated running cost $\tilde{c} : \mathcal{X} \times \mathcal{U} \rightarrow [0, \infty)$ by

$$\tilde{c}(\mathbf{x}, \mathbf{u}) = \sum_{k=0}^{r(\mathbf{x}, \mathbf{u})-1} c(f^k(\mathbf{x}, \mathbf{u}), \mathbf{u}). \quad (4)$$

Note that we can reconstruct the “true time” k from the “event time” ℓ by the event function r : we have that

$$k(\ell + 1) = k(\ell) + r(\mathbf{x}(\ell), \mathbf{u}(\ell)).$$

2. *Quantization model*: The controller only receives *quantized* information on the state. Formally, we are given a (finite) partition $P = \{\mathcal{P}_1, \dots, \mathcal{P}_d\}$, $\mathcal{P}_i \subset \mathcal{X}$, of \mathcal{X} which induces an equivalence relation \sim on $\mathcal{X} \times \mathcal{X}$ by $x \sim y \Leftrightarrow x$ and y lie in the same partition element. We denote by $[\mathbf{x}] \in P$ the corresponding equivalence class of $\mathbf{x} \in \mathcal{X}$. Only $[\mathbf{x}(\ell)]$ is transmitted from the plant to the controller at the event time ℓ . Thus, from the viewpoint of the controller, the plant is given by the finite state system, cf. [GJ08, GM09]

$$\mathcal{P}(\ell + 1) = F(\mathcal{P}(\ell), \mathbf{u}(\ell), \gamma(\ell)), \quad \ell = 0, 1, \dots, \quad (5)$$

defined by

$$F(\mathcal{P}, \mathbf{u}, \gamma) = [\tilde{f}(\gamma(\mathcal{P}), \mathbf{u})], \quad \mathcal{P} \in P, \mathbf{u} \in \mathcal{U},$$

where $\gamma : P \rightarrow \mathcal{X}$ denotes a *choice function* which satisfies $[\gamma(\mathcal{P})] = \mathcal{P}$ for all $\mathcal{P} \in P$. The choice function models the fact that it is unknown to the controller from which exact state $\mathbf{x}(\ell)$ the system transits to the next cell $\mathcal{P}(\ell + 1)$. We let Γ denote the set of those functions. Thus, in each step, the dynamics is influenced by the two control parameters $\mathbf{u}(\ell)$ and $\gamma(\ell)$. Our goal is to choose $\mathbf{u}(\ell)$ in each step such that the state $\mathcal{P}(\ell)$ is controlled into the target set. At the same time, the influence of the choice function may be viewed as a perturbation which might prevent us from reaching \mathcal{X}^* . In this sense, (5) constitutes a *dynamic game*, cf. [GJ08].

2.1 Computing the optimal feedback

In order to be compatible with our quantization, from now on we assume that \mathcal{X}^* is given by the union of some elements from P . For the quantized system (5) we define

$$C(\mathcal{P}, \mathbf{u}) := \sup_{\mathbf{x} \in \mathcal{P}} \tilde{c}(\mathbf{x}, \mathbf{u}).$$

For given $\mathcal{P}(0) \in P$, $\underline{\mathbf{u}} = (\mathbf{u}(\ell))_\ell \in \mathcal{U}^\mathbb{N}$ and $\underline{\gamma} = (\gamma(\ell))_\ell \in \Gamma^\mathbb{N}$, the cost accumulated along the associated trajectory $(\mathcal{P}(\ell))_\ell \in P^\mathbb{N}$ of (5) (which depends on $\underline{\mathbf{u}}$ and $\underline{\gamma}$) is

$$J(\mathcal{P}(0), \underline{\mathbf{u}}, \underline{\gamma}) := \sum_{\ell=0}^L C(\mathcal{P}(\ell), \mathbf{u}(\ell)) \in [0, \infty],$$

where $L = \min\{\ell \geq 0 : \mathcal{P}(\ell) \subset \mathcal{X}^*\} \in \mathbb{N} \cup \{\infty\}$. The (*upper*) *optimal value function* is

$$V(\mathcal{P}) := \sup_{\Gamma} \inf_{\underline{\mathbf{u}} \in \mathcal{U}^\mathbb{N}} J(\mathcal{P}, \underline{\mathbf{u}}, \Gamma(\underline{\mathbf{u}})) \in [0, \infty],$$

where $\Gamma : \mathcal{U}^\mathbb{N} \rightarrow \Gamma^\mathbb{N}$ is a strategy of the form

$$\begin{aligned} \Gamma(\underline{\mathbf{u}}) &= \Gamma((\mathbf{u}(\ell))_\ell) \\ &= (\gamma_1(\mathbf{u}(1)), \gamma_2(\mathbf{u}(1), \mathbf{u}(2)), \gamma_3(\mathbf{u}(1), \mathbf{u}(2), \mathbf{u}(3)), \dots) \end{aligned}$$

and the sup in the definition of the optimal value function is over all strategies of this form. The optimal value function – by standard arguments, cf. [Ber95] – for \mathcal{P} in the stabilizable set $S = \{\mathcal{P} \in P \mid V(\mathcal{P}) < \infty\}$ is the unique solution to the *optimality principle*

$$V(\mathcal{P}) = \inf_{\underline{\mathbf{u}} \in \mathcal{U}} \left\{ C(\mathcal{P}, \mathbf{u}) + \sup_{\gamma \in \Gamma} V(F(\mathcal{P}, \mathbf{u}, \gamma)) \right\} \quad (6)$$

together with the boundary condition $V(\mathcal{P}) = 0$ for $\mathcal{P} \subset \mathcal{X}^*$. Given V , we obtain an optimal feedback for (5) by setting

$$u(\mathcal{P}) := \operatorname{argmin}_{\mathbf{u} \in \mathcal{U}} \left\{ C(\mathcal{P}, \mathbf{u}) + \sup_{\gamma \in \Gamma} V(F(\mathcal{P}, \mathbf{u}, \gamma)) \right\} \quad (7)$$

for $\mathcal{P} \in S$. Note that we also immediately obtain a feedback for the original system (1) resp. (2) from this by setting $u(\mathbf{x}) := u([\mathbf{x}])$ for $\mathbf{x} \in \bigcup_{\mathcal{P} \in S} \mathcal{P}$. By construction, any trajectory of the closed loop system

$$\mathbf{x}(k+1) = f(\mathbf{x}(k), u([\mathbf{x}(k)])) \quad (8)$$

with $\mathbf{x}(0) \in S$ is eventually reaching the target set \mathcal{X}^* , cf. [GJ07, GM09].

Computationally, the feedback is constructed using a directed, weighted hypergraph together with a corresponding shortest path algorithm, cf. [JO04, GJ08, vL07] (see also Section 5).

3 Construction of a lazy feedback

When the data transmission between the plant and the controller is realized via a digital network it is often desirable to minimize the amount of transmitted information in order to reduce the overall network load. More specifically, here we treat the question of how to minimize the number of times that a new control value has to be transmitted from the controller to the plant. Using an optimization based feedback construction, this goal can directly be modelled by suitably defining the running cost function.

In order to detect a change in the control value generated by the controller we need to be able to compare to its value from the previous time step (resp. event). We therefore define the *extended state space* $\mathcal{Z} := \mathcal{X} \times \mathcal{U}$. Based on the event system (2), we consider the event system

$$\mathbf{z}(\ell + 1) = g(\mathbf{z}(\ell), \mathbf{u}(\ell)), \quad \ell = 0, 1, 2, \dots \quad (9)$$

with $\mathbf{z}(\ell) = (\mathbf{x}(\ell), \mathbf{w}(\ell)) \in \mathcal{Z}$ the extended state vector, $\mathbf{u}(\ell) \in \mathcal{U}$ and $g : \mathcal{Z} \times \mathcal{U} \rightarrow \mathcal{Z}$ is defined by

$$g(\mathbf{z}, \mathbf{u}) = g((\mathbf{x}, \mathbf{w}), \mathbf{u}) = \begin{bmatrix} \tilde{f}(\mathbf{x}, \mathbf{u}) \\ \mathbf{u} \end{bmatrix}.$$

We define $\mathcal{Z}^* := \mathcal{X}^* \times \mathcal{U}$ as the target set in the extended state space so that reaching the target only depends on the state \mathbf{x} . We further define an associated running cost function $d : \mathcal{Z} \times \mathcal{U} \rightarrow [0, \infty)$ by

$$d((\mathbf{x}, \mathbf{w}), \mathbf{u}) = (1 - \lambda)\tilde{c}(\mathbf{x}, \mathbf{u}) + \lambda(1 - \delta(\mathbf{u} - \mathbf{w})) \quad (10)$$

with

$$\delta(\mathbf{u}) := \begin{cases} 1, & \text{if } \mathbf{u} = 0, \\ 0, & \text{else.} \end{cases} \quad (11)$$

Here, $\lambda \in [0, 1)$ must be strictly < 1 in order to guarantee that $d(\mathbf{z}, \mathbf{u}) = 0$ iff $\mathbf{z} \in \mathcal{Z}^*$. Note that the dynamics of the extended system (9) does not depend on the second component \mathbf{w} of the extended state vector \mathbf{z} , but only the modified cost function d does.

We can now apply the construction from the previous section to the system (9) with cost function (10). To this end, we would need to construct a partition of \mathcal{U} . Instead, in order to simplify the exposition, here we simply assume that \mathcal{U} is discrete, i.e. contains only finitely many elements. We then use $P \times \mathcal{U}$ as the underlying partition for the quantization. We denote the resulting optimal value function by $V_\lambda : \mathcal{Z} \rightarrow [0, \infty]$, the stabilizable subset by

$$S_\lambda := \{\mathbf{z} \in \mathcal{Z} : V_\lambda([\mathbf{z}]) < \infty\}$$

and the associated feedback by $u_\lambda : S_\lambda \rightarrow \mathcal{U}$.

We will show that for a sufficiently large $\lambda < 1$ the closed loop system

$$\mathbf{z}(\ell + 1) = g(\mathbf{z}(\ell), u_\lambda(\mathbf{z}(\ell))), \quad \ell = 0, 1, 2, \dots, \quad (12)$$

is asymptotically stable on S_λ , i.e. that for $\mathbf{z}(0) \in S_\lambda$ the associated trajectory enters \mathcal{Z}^* in finitely many steps. Furthermore, the number of control value changes along this trajectory will be minimal.

To be more precise: For some initial state $\mathbf{z} = \mathbf{z}(0) \in S_\lambda$ let $(\mathbf{z}(\ell))_\ell \in \mathcal{Z}^\mathbb{N}$, $\mathbf{z}(\ell) = (\mathbf{x}(\ell), \mathbf{w}(\ell))$, be the corresponding trajectory of the closed loop system (12), let

$$L(\mathbf{z}, u_\lambda) = \min\{\ell \geq 0 : \mathbf{z}(\ell) \in \mathcal{Z}^*\}$$

be the number of time steps until the trajectory reaches the target set \mathcal{Z}^* ,

$$E(\mathbf{z}, u_\lambda) = \sum_{\ell=0}^{L(\mathbf{z}, u_\lambda)} 1 - \delta(u_\lambda(\mathbf{z}(\ell)) - \mathbf{w}(\ell))$$

the number of control value changes along the corresponding trajectory as well as

$$\tilde{J}(\mathbf{z}, u_\lambda) = \sum_{\ell=0}^{L(\mathbf{z}, u_\lambda)} \tilde{c}(\mathbf{x}(\ell), u_\lambda(\mathbf{z}(\ell)))$$

the accumulated (original) costs.

Theorem 1. *For all $\lambda \in [0, 1)$, $S \times \mathcal{U} \subset S_\lambda$ and $\mathbf{x}(\ell) \rightarrow \mathcal{X}^*$ as $\ell \rightarrow \infty$.*

Further, there exists a $\lambda < 1$ such that for any feedback $u : S_\lambda \rightarrow \mathcal{U}$ for the extended system and $\mathbf{z} \in S_\lambda$ with $L(\mathbf{z}, u) < \infty$ holds $E(\mathbf{z}, u) \geq E(\mathbf{z}, u_\lambda)$.

Proof. By definition, the extended system (9) and the cost function (10) fulfill the assumptions in [GJ07], so asymptotic stability of the closed loop system (12) directly follows for all $\mathbf{z} \in S_\lambda$ by their proof.

In order to show that $S \times \mathcal{U} \subset S_\lambda$ for all $\lambda \in [0, 1)$, choose $\lambda \in [0, 1)$ and some initial value $\mathbf{z}(0) = (\mathbf{x}(0), \mathbf{u}(0)) \in S \times \mathcal{U}$ arbitrarily. Consider the feedback

$$u(\mathbf{z}) = u((\mathbf{x}, \mathbf{u})) := u(\mathbf{x})$$

for system (9), where $u(\mathbf{x})$ denotes the feedback for (2) which has been constructed in Section 2. This leads to a trajectory $(\mathbf{x}(\ell), \mathbf{u}(\ell))_\ell$ of the extended system with $(\mathbf{x}(\ell))_\ell$ being exactly the trajectory of the original system (2). Since $\mathbf{x}(0) \in S$, $V(\mathbf{x}(0))$ is finite and the accumulated cost $\tilde{J}(\mathbf{z}(0), u)$ for this trajectory does not exceed $(1 - \lambda)V(\mathbf{x}(0)) + \lambda L(\mathbf{z}(0), u)$ which is finite. According to the optimality of V_λ ,

$$V_\lambda(\mathbf{z}(0)) \leq (1 - \lambda)V(\mathbf{x}(0)) + \lambda L(\mathbf{z}(0), u) < \infty$$

follows, i.e. $\mathbf{z}(0) \in S_\lambda$.

To show the optimality of u_λ with respect to the number of control value changes, assume there exists a feedback $\bar{u} : S_\lambda \rightarrow \mathcal{U}$ with $E(\mathbf{z}, \bar{u}) \leq E(\mathbf{z}, u_\lambda) - 1$ for some $\mathbf{z} \in S_\lambda$. Since u_λ is optimal, the following inequality holds:

$$(1 - \lambda)\tilde{J}(\mathbf{z}, \bar{u}) + \lambda E(\mathbf{z}, \bar{u}) \tag{13}$$

$$\geq (1 - \lambda)\tilde{J}(\mathbf{z}, u_\lambda) + \lambda E(\mathbf{z}, u_\lambda) \tag{14}$$

$$\geq (1 - \lambda)\tilde{J}(\mathbf{z}, u_\lambda) + \lambda E(\mathbf{z}, \bar{u}) + \lambda \tag{15}$$

and thus

$$(1 - \lambda)\tilde{J}(\mathbf{z}, \bar{u}) \geq (1 - \lambda)\tilde{J}(\mathbf{z}, u_\lambda) + \lambda.$$

As the costs denoted by \tilde{J} are finite, $\lambda \rightarrow 1$ leads to a contradiction. \square

4 Numerical experiments

4.1 Nonlinear Inverted Pendulum

For our numerical experiments we first consider an inverted pendulum on a cart, cf. [JYH01]. The motion of the pendulum is given by the continuous time control system

$$\left(\frac{4}{3} - m_r \cos^2 \varphi\right) \ddot{\varphi} + \frac{m_r}{2} \dot{\varphi}^2 \sin 2\varphi - \frac{g}{\ell} \sin \varphi = -u \frac{m_r}{m\ell} \cos \varphi,$$

where $(\varphi, \dot{\varphi}) \in [0, 2\pi] \times \mathbb{R}$ denotes the state of the pendulum and $u \in \mathcal{U} \subset \mathbb{R}$ is the control input. We have used the parameters $m = 2$ for the pendulum mass, $m_r = m/(m + M)$ for the mass ratio with cart mass $M = 8$, $\ell = 0.5$ as the length of the pendulum and $g = 9.8$ for the gravitational constant. As the instantaneous cost function, we employ

$$q(\varphi, \dot{\varphi}, u, t) = \frac{0.01}{2} u^2 + t, \tag{16}$$

where $t \in \mathbb{R}$ is the system's time. Denoting the system's evolution operator for constant control functions $u(t) \equiv \mathbf{u} \in \mathcal{U}$ by $\Phi^t(\mathbf{x}, \mathbf{u})$, $\mathbf{x} = (\varphi, \dot{\varphi})$, we consider the discrete time system $f(\mathbf{x}, \mathbf{u}) = \Phi^T(\mathbf{x}, \mathbf{u})$ for $T = 0.01$, i.e., the sampled continuous time system with sampling rate $T = 0.01$. The map Φ^T is approximated via the classical Runge-Kutta scheme of order 4 with 5 equidistant steps. The discrete time cost function is obtained by numerically integrating the continuous time instantaneous cost according to $c(\mathbf{x}, \mathbf{u}) = \int_0^T q(\Phi^t(\mathbf{x}, \mathbf{u}), \mathbf{u}, t) dt$. We choose the state space $\mathcal{X} = [-10, 10] \times [-8, 8]$ and a partition of $2^7 \times 2^7$ equally sized partition elements on \mathcal{X} . The control space is chosen as $\mathcal{U} = [-64, 64]$, discretized by 17 equidistant samples and the target region is set to $[-\frac{5}{8}, \frac{5}{8}] \times [-\frac{1}{2}, \frac{1}{2}]$ (i.e. 8×8 partition elements around $[0, 0]$).

Let $\mathbf{s}(\mathbf{x}) \in \mathbb{R}^2$ and $\mathbf{t}(\mathbf{x}) \in \mathbb{R}^2$ denote the center and the radius of the rectangular partition element containing \mathbf{x} , respectively, then we define an *event set* via

$$\beta(\mathbf{x}) = \{\mathbf{y} = (y_1, y_2)^T \in \mathcal{X} : |y_i - s_i(\mathbf{x})| \leq e_r t_i(\mathbf{x}), i = 1, 2\} \quad (17)$$

with *event radius* $e_r = 9$ and the event function

$$r(\mathbf{x}, \mathbf{u}) = \begin{cases} \min\{t \in 0.01\mathbb{N} : \Phi^t(\mathbf{x}, \mathbf{u}) \notin \beta(\mathbf{x})\}, \\ \text{if not empty} \\ 0, \text{ else.} \end{cases} \quad (18)$$

For this quantized event system, we employ the construction in the two previous sections,

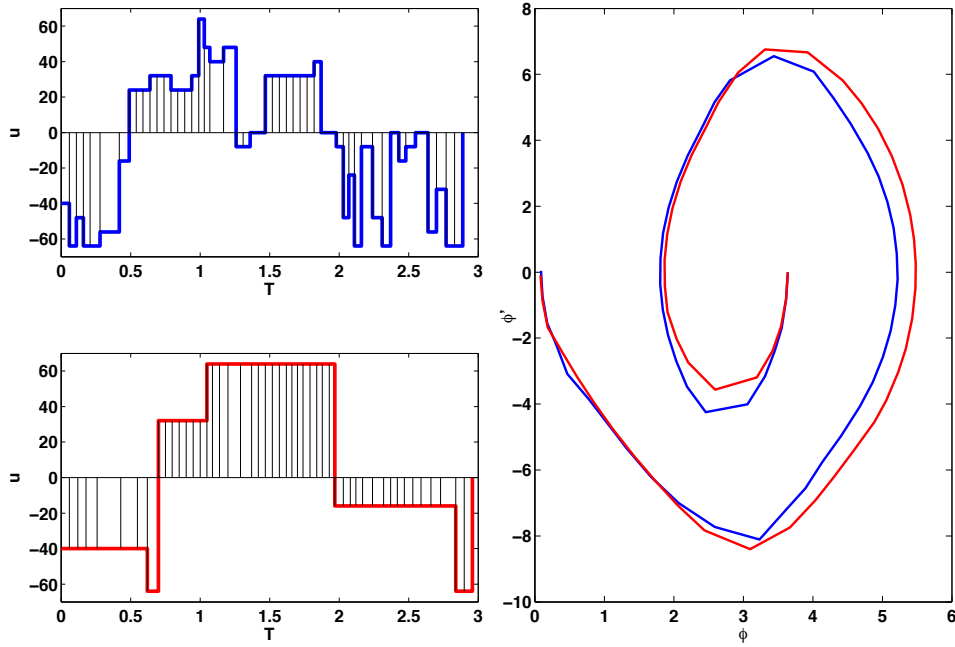


Figure 1: Inverted pendulum: Control sequence over time (left) and the associated trajectories in state space for the initial state $(\pi + 0.5, 0)^T$. Blue (dark): generated by the ordinary feedback, red (light): generated by the lazy feedback.

i.e. we compute the (“ordinary”) feedback as described in Section 2 as well as the lazy feedback from Section 3 with $\lambda = 0.99$.

Figure 1 shows two trajectories of the closed loop system (8) starting at the initial state $(\pi + 0.5, 0)^T \in \mathcal{X}$. On the right-hand side we show these two trajectories in state space while on the left-hand side we plot the associated control sequences over time. The blue (dark) trajectory results from the ordinary feedback, while the red (light) trajectory is generated with the lazy feedback.

The gray ticks on the left indicate all time instances where an event occurs. While the ordinary feedback leads to a change of the control value at almost every event and generates a total number of 32 control value changes, the lazy feedback stabilizes the system using only 5 control changes. Note that the time until the target set is reached (which is essentially minimized here) remains almost the same.

4.2 Batch Reactor

In this numerical experiment, the aim is to control a thermofluid process in a batch reactor (cf. Figure 2) as described in [GJJ⁺10]. The main part of the process consists in the cylindrical batch reactor TB which has a continuously adjustable inflow via valve V_1 of water from the spherical tank T3 above. In addition, a permanent outflow only depending on the fluid level in TB is present. Heating rods can increase the temperature

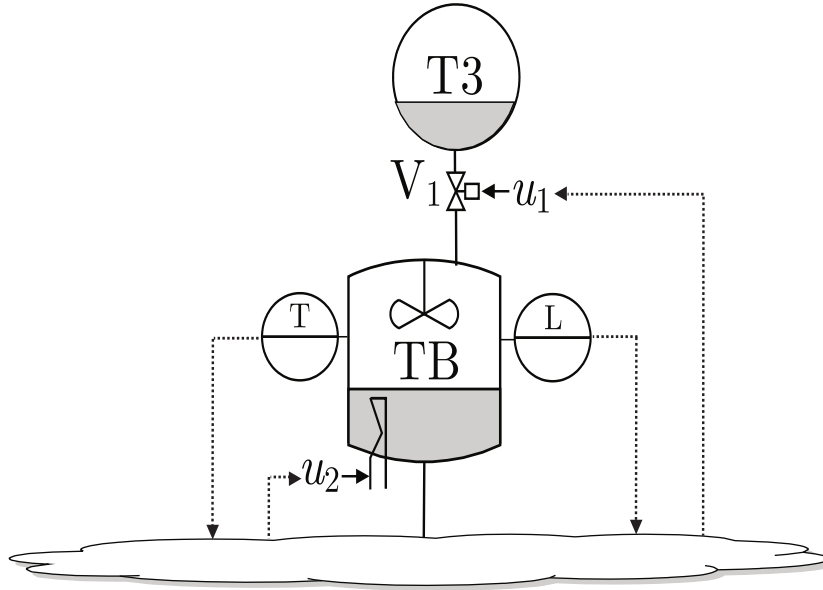


Figure 2: Thermofluid process

of the fluid in TB while cooling can only be achieved by the inflow of cool water from T3. The two state variables that can be continuously measured are the fluid level l_{TB} and the temperature ϑ_{TB} in TB, so the state becomes $\mathbf{x} = (x_1, x_2) = (l_{TB}, \vartheta_{TB})$. The valve angle $u_1 \in [0, 1]$ of valve V_1 and the power $u_2 \in \{0, \dots, 6\}$ of the heating rods are considered as input $\mathbf{u} = (u_1, u_2)$. To model the nonlinear dynamics of the process we use the following differential equation with parameters from Table 1 (cf. [GJJ⁺10]):

$$\begin{aligned}\dot{x}_1 &= \frac{1}{A_h} \left(q_{T3}(u_1) - K_A \sqrt{2gx_1} \right) \\ \dot{x}_2 &= \frac{1}{V(x_1)} \left(q_{T3}(u_1)(\vartheta_{T3} - x_2) + \frac{P_{el}k_h u_2}{\varrho c_p} \right),\end{aligned}\tag{19}$$

where

$$\begin{aligned}q_{T3}(u) &= \begin{cases} 7 \cdot 10^{-6}(11.1u^2 + 13.1u + 0.2) & \text{for } u > 0.2, \\ 0 & \text{else} \end{cases} \\ V(x) &= 0.07x - 1.9 \cdot 10^{-3} \text{ for } x > 0.26.\end{aligned}\tag{20}$$

The unit of the flow q_{T3} is m^3/s and the unit of the volume V of the fluid in TB is m^3 .

Table 1: Parameters and constants

Parameter	Value	Meaning
P_{el}	3000 W	Electrical power
k_h	0.84 J/(W s)	Heat transfer coefficient
c_p	4180 J/(kg K)	Heat capacity of water
g	9.81 m/s ²	Gravitation constant
ϱ	998 kg/m ³	Density of water
ϑ_{T3}	293.15 K	Temperature of inflow
K_A	$1.59 \cdot 10^{-5} \text{ m}^3/\text{m}$	Outflow parameter
A_h	0.07m ²	Cross sectional area

The goal is to steer the system into a neighbourhood of the operating point $(\bar{l}_{TB}, \bar{\vartheta}_{TB}) = (0.349 \text{ m}, 310.56 \text{ K})$ with only a minimum number of control changes, therefore we use a weighted sum of time and a quadratic function in \mathbf{x} and \mathbf{u} with a small gain as cost function. According to the physical limitations of the reactor, we set the state space to $\mathcal{X} = [0.26, 0.45] \text{ m} \times [293.15, 323.15] \text{ K}$ and use a discretization of $2^6 \times 2^6$ partition elements. The continuous input $u_1 \in [0, 1]$ is discretized via 12 equidistant samples. For the time integration of the ordinary differential equation, we use the classical Runge-Kutta scheme of order 4 with 5 equidistant steps and a time step of 1s. An event is generated whenever the state leaves a partition element, i.e. we employ an event radius of $e_r = 1$ here.

We consider the initial state $[0.275 \text{ m}, 295 \text{ K}]$ and – like in the first example – compare two trajectories associated, respectively, to the ordinary feedback described in Section 2 and the lazy one as described in Section 3 with $\lambda = 0.9$. In Figure 3 (left) we compare the generated control sequences. Using the lazy feedback, the number of control value changes is reduced dramatically from 13 to only 2. Note that, again, the time required for the system to reach the target set remains almost the same, cf. the right part of Figure 3.

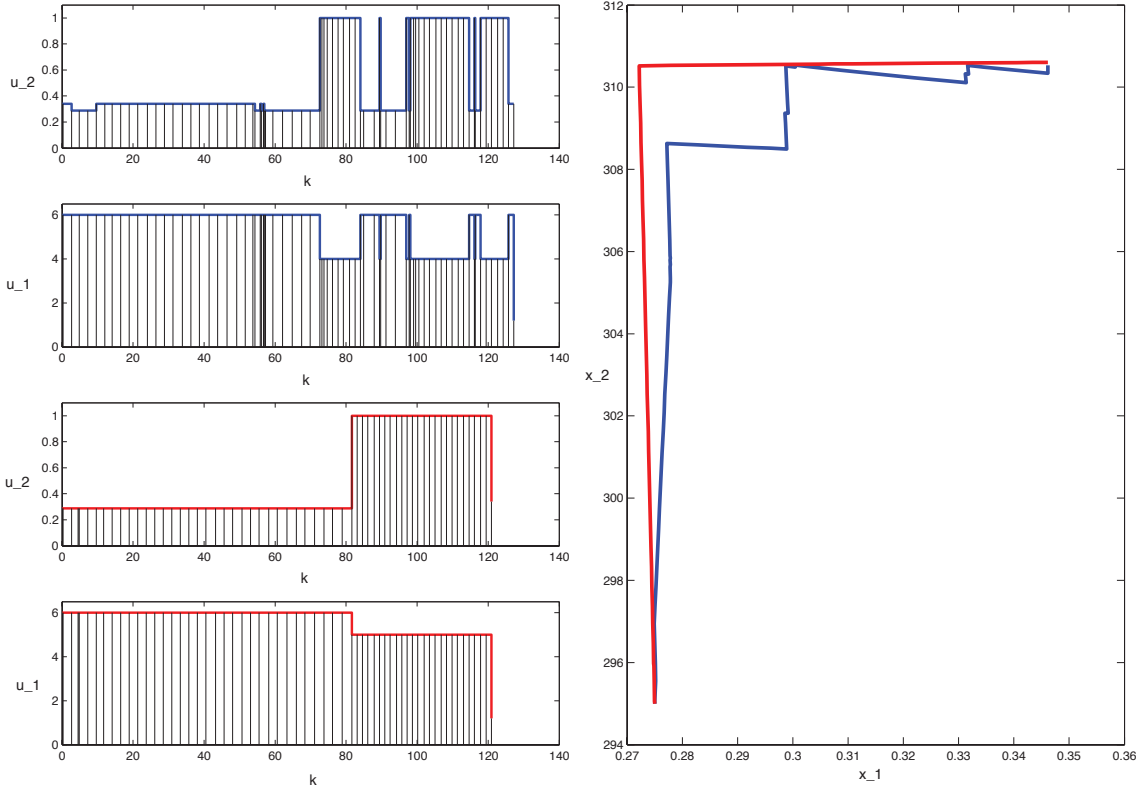


Figure 3: Left: Output of the controller over time for the batch process with initial state $[0.275m, 295K]$. Upper figure: using the standard cost function. Lower figure: lazy feedback, i.e. minimal number of control value changes ($\lambda = 0.9$). Right: Associated feedback trajectories in state space. Blue (dark): using the standard cost function. Red (light): lazy feedback, i.e. minimal number of control value changes ($\lambda = 0.9$).

5 A heuristic approach for reducing the data transmission frequency

One drawback of the lazy feedback construction proposed in Section 3 is the need to extend the state space to $\mathcal{X} \times \mathcal{U}$. This leads to a notable increase in both memory requirements and computing time. In this section we will illustrate that this expansion is really needed in order to obtain the *minimum* number of control changes. Nevertheless, there is a heuristic way of *reducing* the number of control changes without these drawbacks. In order to derive this heuristic, we first have a closer look at the graph construction which is used in order to compute the value function and the feedback, cf. [JO04, GJ08].

For each partition element $\mathcal{P} \in P$, the value $V(\mathcal{P})$ is given by the length of the

shortest path from \mathcal{P} to the set of partition elements which constitute \mathcal{X}^* in the following hypergraph $G = (P, \mathcal{E})$: The edges of G are given by

$$\mathcal{E} = \{(\mathcal{P}, F(\mathcal{P}, \mathbf{u}, \Gamma)) \mid \mathcal{P} \in P, \mathbf{u} \in \mathcal{U}\}$$

weighted by

$$w(\mathcal{P}, N) = \inf\{C(\mathcal{P}, \mathbf{u}) : \mathbf{u} \in \mathcal{U}, F(\mathcal{P}, \mathbf{u}, \Gamma) = N\},$$

where $F(\mathcal{P}, \mathbf{u}, \Gamma) := \{F(\mathcal{P}, \mathbf{u}, \gamma) \in P \mid \gamma \in \Gamma\}$. As such, it can be computed by an efficient Dijkstra-type algorithm [GJ07, vL07], cf. Algorithm 1.

Algorithm 1 MinMax-Dijkstra

Input: hypergraph (P, \mathcal{E})
weights $w : \mathcal{E} \rightarrow (0, \infty)$
set of target nodes $O \subset P$
Output: value function $V : P \rightarrow [0, \infty]$
control input $u : P \rightarrow \mathcal{U}$

```

1: for all  $\mathcal{P} \in P \setminus O$  do
2:    $V(\mathcal{P}) := \infty$ 
3: for all  $\mathcal{P} \in O$  do
4:    $V(\mathcal{P}) := 0$ 
5:  $Q := P$ 
6: while  $Q \neq \emptyset$  do
7:    $\mathcal{P} := \operatorname{argmin}_{\mathcal{P}' \in Q} V(\mathcal{P}')$ 
8:    $Q := Q \setminus \{\mathcal{P}\}$ 
9:   for all  $(\mathcal{Q}, N) \in \mathcal{E}$  with  $\mathcal{P} \in N$  do
10:    if  $N \subset P \setminus Q$  then
11:      if  $V(\mathcal{Q}) > w(\mathcal{Q}, N) + V(\mathcal{P})$  then
12:         $V(\mathcal{Q}) := w(\mathcal{Q}, N) + V(\mathcal{P})$ 
13:         $u(\mathcal{Q}) := u(\mathcal{Q}, N)$ 

```

Here, $u(\mathcal{Q}, N) = \operatorname{argmin}_{\mathbf{u} \in \mathcal{U}}\{C(\mathcal{P}, \mathbf{u}) \mid F(\mathcal{Q}, \mathbf{u}, \Gamma) = N\}$. One easily shows that when a node \mathcal{P} is removed from Q in line 6, $V(\mathcal{P})$ and $u(\mathcal{P})$ are fixed until termination (cf. for example [AMOT90]). It follows that when a hyperedge (\mathcal{Q}, N) is being processed in lines 9 and 10, the value $V(\mathcal{P})$ is fixed for all target nodes $\mathcal{P} \in N$ of the hyperedge because $N \subset P \setminus Q$ (line 7, 8). This information can be used for the choice of a proper control $u(\mathcal{Q})$.

To be more precise, let $u(\mathcal{Q}, N)$ denote the control applied for hyperedge (\mathcal{Q}, N) , $u_0(\mathcal{P})$ an arbitrary control for the target nodes $\mathcal{P} \in O$ and $\lambda \in [0, 1)$ a parameter. Then in order to reduce the number of control changes we change lines 3 – 4 and lines 11 – 13 in Algorithm 1 as follows:

```

3: for all  $\mathcal{P} \in O$  do
4:    $V(\mathcal{P}) := 0$  and  $u(\mathcal{P}) := u_0(\mathcal{P})$ 

11: if  $V(\mathcal{Q}) > (1 - \lambda)w(\mathcal{Q}, N) + \lambda\sigma(u(\mathcal{Q}, N), N) + V(\mathcal{P})$  then
12:    $V(\mathcal{Q}) := (1 - \lambda)w(\mathcal{Q}, N) + \lambda\sigma(u(\mathcal{Q}, N), N) + V(\mathcal{P})$ 
13:    $u(\mathcal{Q}) := u(\mathcal{Q}, N)$ 

```

Here, $\sigma : \mathcal{U} \times P^{[N]} \rightarrow [0, \infty)$ is a function dependent on the control of the hyperedge and the controls of the possible subsequent states. In order to reduce the number of control changes along trajectories, $\sigma(u, N)$ has to be large when $u \neq u(\mathcal{P})$ for many $\mathcal{P} \in N$ and should be small otherwise. For example, one can set

$$\sigma(u, N) = 1 - \delta \left(u \left(\underset{\mathcal{P} \in N}{\operatorname{argmax}} V(\mathcal{P}) \right) - u(\mathcal{Q}, N) \right)$$

with δ defined in (11). In our numerical tests, however, choosing

$$\sigma(u, N) = \frac{1}{|N|} \sum_{\mathcal{P} \in N} 1 - \delta(u(\mathcal{P}) - u(\mathcal{Q}, N))$$

seemed to lead to better results.

5.1 A counterexample

In order to show that the heuristic approach in general does not lead to a feedback which produces the minimal number of control value changes, we consider the following counterexample, cf. Figure 4.

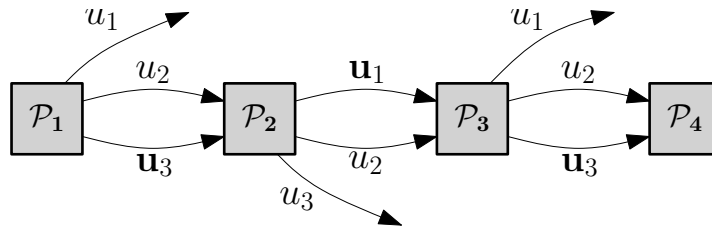


Figure 4: Counterexample to illustrate that the heuristic approach in general does not lead to a feedback which minimizes the number of control changes

We start with node \mathcal{P}_4 on the right and assume that control input u_1 is optimal for node \mathcal{P}_4 . Let \mathcal{P}_3 be the neighbour of \mathcal{P}_4 being processed next in Algorithm 1. As control input u_1 does not lead to a transition to \mathcal{P}_4 we only have the choice between u_2 and u_3 . Since both of them lead to a control change, we may chose as well u_3 . This

procedure can be done recursively up to state \mathcal{P}_1 on the left with u_1 and u_2 or u_1 and u_3 , respectively. We end up with a trajectory from \mathcal{P}_1 to \mathcal{P}_4 switching control 3 times, which is not the optimum compared to having constant control u_2 and switching control only once at the end.

The reason for this effect is that for each node \mathcal{P} we only have computed the optimal value $V(\mathcal{P})$ and control input $u(\mathcal{P})$ with respect to a trajectory that starts at \mathcal{P} . But we do not know about optimality when \mathcal{P} is not the initial state of a trajectory. A list of the values $V(\mathcal{P})$ for all pairs $(\mathcal{P}, \mathbf{u}) \in P \times \mathcal{U}$ would be sufficient to remedy this – but this then leads directly back to the lazy feedback approach of section 3.

5.2 Numerical experiments

Nevertheless, in numerical experiments we often obtain reasonable results even with the heuristic approach. The number of control changes often may be reduced without enlarging the state space and so without any noticeable effect on the memory consumption (cf. Figure 5, where the number of control value changes was reduced from 33 to 18).

On the other hand, the heuristic approach does not seem to be prone to failure, as shown in repeating the numerical experiment with the batch reactor. Here (cf. Figure 6), the number of control value changes is actually increased. A closer inspection seems to reveal a phenomenon similar to the counterexample described above.

6 Conclusion

Based on an extended model of a nonlinear quantized event system we introduced the concept of a lazy feedback which stabilizes the system with the minimal number of control value changes. We show that the lazy feedback is indeed optimal in this sense among all feedbacks which stabilize the same set of initial conditons. In addition, we illustrated a heuristic method which sometimes can be used for reducing the number of control changes without any further computational effort compared to the lazy feedback construction.

References

- [AB02] K. Aström and B. Bernhardsson. Comparison of Riemann and Lebesgue sampling for first order stochastic systems. In *Proc. of IEEE Conf. on Decision and Control*, volume 2, pages 2011–2016, Las Vegas, USA, 2002.
- [AMOT90] R.K. Ahuja, K. Mehlhorn, J. Orlin, and R.E. Tarjan. Faster algorithms for the shortest path problem. *J. ACM*, 37(2):213–223, 1990.

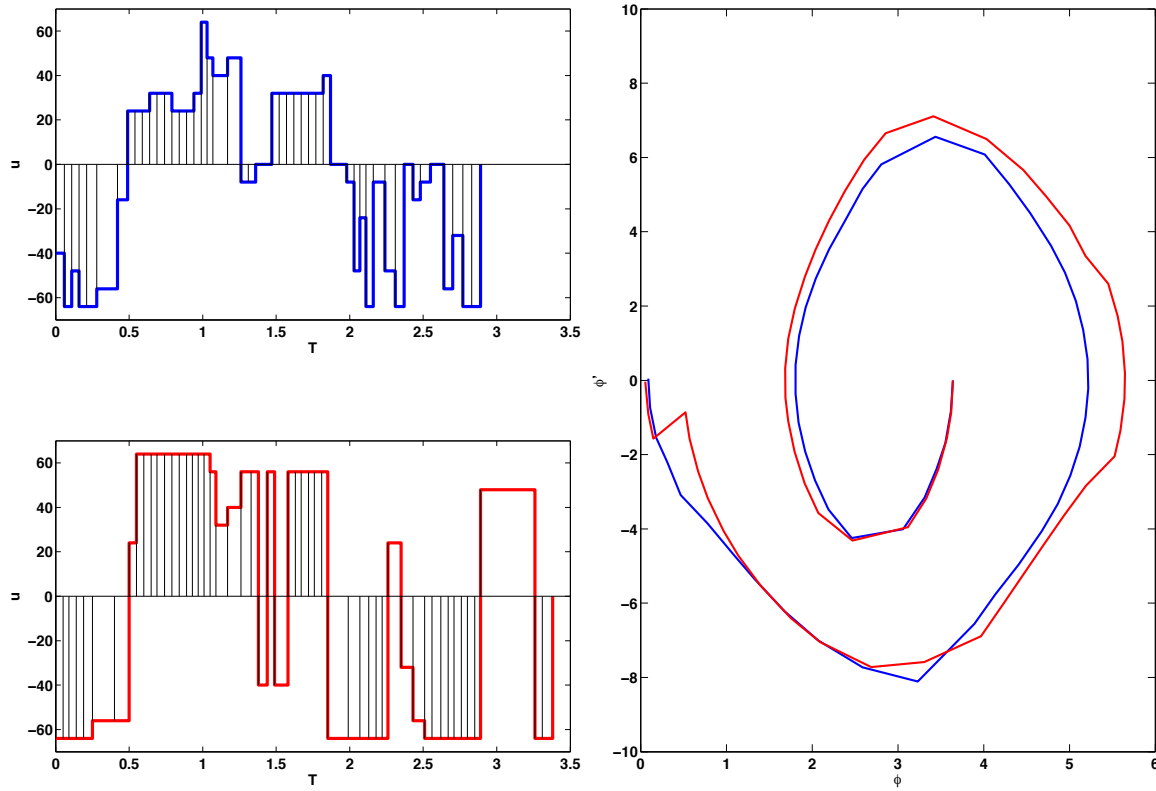


Figure 5: Inverted pendulum: Control sequences (left) and state space trajectory (right) for the ordinary feedback (blue/dark) in comparison to the heuristic approach for reducing the number of control changes (red/light). Parameters are the same as in Section 4.

- [Arz99] K. Arzen. A simple event-based PID controller. In *Proceedings of IFAC World Congress*, pages 423–428, 1999.
- [Ast08] K. Aström. Event based control. In A. Astolfi and L. Marconi, editors, *Analysis and Design of Nonlinear Control Systems*, pages 127–147. Springer-Verlag, 2008.
- [AW97] K. J. Aström and B. Wittenmark. *Computer-Controlled Systems*. Prentice Hall, 1997.
- [Ber95] D. P. Bertsekas. *Dynamic Programming and Optimal Control. Vol. 2*. Belmont, MA: Athena Scientific, 1995.
- [FJL02] D. Förstner, M. Jung, and J. Lunze. A discrete-event model of asynchronous quantised systems. *Automatica*, 38(8):1277–1286, 2002.

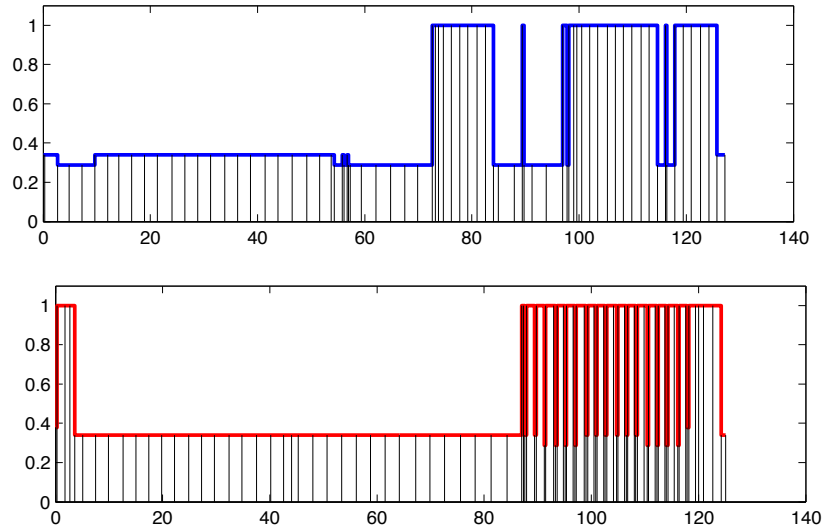


Figure 6: Batch reactor: Control sequences (left) and state space trajectory (right) for the ordinary feedback (blue/dark) in comparison to the heuristic approach for reducing the number of control changes (red/light). Parameters are the same as in Section 4.

- [GJ07] L. Grüne and O. Junge. Approximately optimal nonlinear stabilization with preservation of the Lyapunov function property. In *Proc. of the 46th IEEE CDC*, pages 702–707, 2007.
- [GJ08] L. Grüne and O. Junge. Global optimal control of perturbed systems. *J. Optim. Theory Appl.*, 136(3):411–429, 2008.
- [GJJ⁺10] L. Grüne, S. Jerg, O. Junge, D. Lehmann, J. Lunze, F. Müller, and M. Post. Two complementary approaches to event-based control. *at – Automatisierungstechnik*, 58(4):173–181, 2010.
- [GM09] L. Grüne and F. Müller. An algorithm for event-based optimal feedback control. In *Proc. of the 48th IEEE CDC*, pages 5311 – 5316, 2009.
- [Hsu92] H. Hsu. Global analysis by cell mapping. *Int. J. Bif. Chaos*, 2:727–771, 1992.
- [HSVDB07] W.P.M.H. Heemels, J. Sandee, and P.P.J. Van Den Bosch. Analysis of event-driven controllers for linear systems. *Int. J. Cont.*, 81(4):571–590, 2007.
- [JO04] O. Junge and H.M. Osinga. A set oriented approach to global optimal control. *ESAIM Control Optim. Calc. Var.*, 10(2):259–270, 2004.

- [JYH01] A. Jadbabaie, J. Yu, and J. Hauser. Unconstrained receding-horizon control of nonlinear systems. *IEEE Trans. Automat. Control*, 46(5):776–783, 2001.
- [KB06] E. Kofman and J.H. Braslavsky. Level crossing sampling in feedback stabilization under data-rate constraints. In *Proc. of IEEE CDC*, pages 4423–4428, 2006.
- [LL09] J. Lunze and D. Lehmann. A state feedback approach to event-based control. *Automatica*, 46(1):211–215, 2009.
- [Lun94] J. Lunze. Qualitative modelling of linear dynamical systems with quantized state measurements. *Automatica*, 30(3):417–431, 1994.
- [OMT02] P.G. Otanez, J.G. Moyne, and D.M. Tilbury. Using deadbands to reduce communication in networked control systems. In *Proc. of ACC*, pages 3015–3020, 2002.
- [Sch03] J. Schroeder. *Modeling, State Observation and Diagnosis of Quantized Systems*. Springer, Berlin, 2003.
- [TW06] P. Tabuada and Xiaofeng Wang. Preliminary results on state-triggered scheduling of stabilizing control tasks. In *Proc. 45th IEEE CDC*, pages 282 – 287, 2006.
- [VK06] V Vasyutynskyy and K Kabitzsch. Implementation of PID controller with send-on-delta sampling. In *Proceedings ICC*, 2006.
- [vL07] Marcus von Lossow. A min-max version of Dijkstra’s algorithm with application to perturbed optimal control problems. *PAMM*, 7(1):4130027–4130028, 2007.